Discovering the City: Crowdsourcing and Personalized Urban Paths Across Cultural Heritage

Giovanni Delnevo[3], Andrea Melis[1(✉)], Silvia Mirri[2], Lorenzo Monti[3], and Marco Prandini[2]

[1] Department of Electrical, Electronic, and Information Engineering "Guglielmo Marconi", Universitá di Bologna, Bologna, Italy
a.melis@unibo.it
[2] Department of Computer Science and Engineering, Universitá di Bologna, Bologna, Italy
[3] Interdepartmental Centre for Industrial ICT Research - CIRI ICT, Universitá di Bologna, Bologna, Italy

Abstract. Travel planners and mobile applications related to cultural heritage can play an interesting role in the development of smart cities, when they are integrated each other, engaging the user in touristic and entertainment activities, letting him/her be a source of cultural resources. This paper focuses on a microservices-based architecture, defined with the aim of providing support in computing personalized urban paths across cultural heritage places and in sharing multimedia resources about points of interest. A prototype of mobile application has been implemented on the basis of such architecture, showing the feasibility of the proposed approach thanks to personas and related scenarios.

Keywords: Smart city · Cultural heritage · Crowdsourcing Microservices · Personal travel planner

1 Introduction

The wide diffusion of smart objects (including smartphones, tablets, smart watches) is profoundly affecting our daily life, changing our habits and the way we conduct all our activities, from communicating to learning, from working to spending free time, and so on. In this context, urban infrastructures, connectivity, and Internet of Things are supporting and enhancing the application and the achievement of smart city paradigms [1]. As a consequence, we are witnessing the proliferation of several mobile applications devoted to support the users in all those activities in the smart city context. In such a scenario, a key role is played by mapping services (e.g. Google Maps) and travel planners (e.g. OpenTripPlanner, Graphhopper), which support users in exploiting urban paths computed by similar algorithms, on the basis of similar characteristics (i.e. time,

B. Guidi et al. (Eds.): GOODTECHS 2017, LNICST 233, pp. 132–141, 2018.
https://doi.org/10.1007/978-3-319-76111-4_14

distance, means of transport, etc.). Generally, such kind of applications shows some lacks in terms of personalization, in fact, the user cannot set any preferences in terms of health and fitness issues, personal and street safety, air quality (pollution, dust and pollen, etc.), user interface and interaction (which can be strongly affected by the use of context [2]), and so on.

Another interesting topic which can get great benefits from exploiting mobile applications on smart devices and from user's preferences mechanisms is the cultural heritage one, related to touristic activities. In particular, geolocalization sensors can play a strategic role in this context, enhancing available functionalities and services [3]. Moreover, in the cultural heritage context, information collected from tourists and citizens [4,5] by means of their mobile devices [6], also thanks to their activities on social networks, could be integrated with the data coming from the official sources (such as data coming from municipalities, touristic offices, museums, public administrations and institutions, private foundations, etc.), enriching the whole amount of available resources and knowledge [7,8]. Again, personalization on the basis of users' preferences could offer great benefits in this field.

This work aims to integrate these two realms, by defining a system architecture based on microservices [9]. In particular, in this paper we focus our attention on how to define and to manage users' preferences, detailing the Preferences Layer of a microservices-based architecture. On the basis of such an architecture, we have designed and developed a prototype of mobile application (named Cicerone) devoted to compute personalized urban paths, so as to let the user move across preferred cultural heritage places and Points Of Interest (POIs) in the city. Such an application has been thought to let the user enjoy multimedia resources about the points of interest s/he is visiting, providing additional data (i.e. pictures, comments, reviews), enriching the information available for those POIs, thanks to crowdsourcing activities and to gamification mechanisms. This work has been done within the SACHER project (Smart Architecture for Cultural Heritage in Emilia Romagna[1], which is co-funded by the Emilia-Romagna Region through the POR FESR 2014-2020 fund - European Regional Development Fund). Hence, we have based our prototype on official data sources about cultural heritage in the city of Bologna and in the Emilia-Romagna Region.

The remainder of the paper is structured as follows. Section 2 provides a description of the Preferences Layer Architecture. Section 3 presents a prototype of mobile application (named Cicerone), implemented on the basis of such an architecture, describing two specific personas and related usage scenarios. Finally, Sect. 4 concludes the paper highlighting some final remarks and future work.

2 The Preferences Layer Architecture

Figure 1 shows the proposed preferences layer architecture. The microservice paradigm is at the basis of the design of our architecture, letting it work on

[1] http://www.eng.sacherproject.com/.

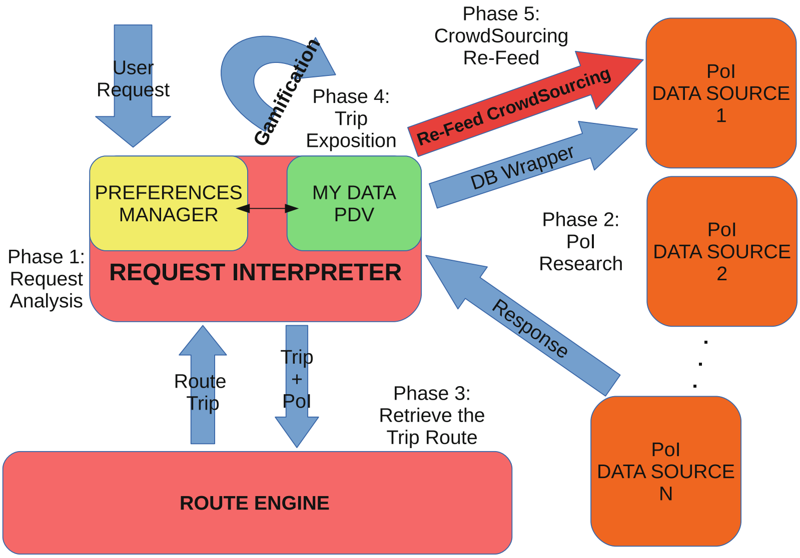

Fig. 1. Preference layer architecture

different and distinct phases [10]. The idea behind such a paradigm is that any function, that can be seen and managed as an independent service, have to be considered as a single microservice. A more complete service will be the result of a proper orchestration of those microservices.

The main advantage of this solution is that we can obtain an extremely modular and distributed architecture [9,11]. This kind of distributed compositionality becomes its greatest strength. As described in [12], even the internal components of the software are autonomous services, independent components conceptually deployed alone with dedicated technologies, both software and hardware. This means that this architectural style does not foster or forbid any particular programming, but it provides a partition structure of the component the developer has to follow. Since all the components of a microservice architecture are microservices, its distinguishing behavior derives from the composition and coordination of its components via messages. Thus, the core of a microservice platform will be the orchestration phase, namely the composition of microservices, tools, and processes invoked, and the connection and automation of workflows that create the final service [13]. However, the microservice paradigm introduces also new challenges and enlarges known threats. The distributed programming nature is at the same time its main advantage and disadvantage [14]. However it has been choose as development paradigm for the ease with which it was possible to disassemble the various stages in independent services. In this way, it could be potentially used later for other architectures. This preference layer interferes between the typical request of a common travel planner (e.g. OTP Analytics) and its response, intercepting the call and editing it based on a user preference processing. The architecture is structured around the orchestration of 5 main phases, which will be detailed in the following subsections:

1. Request and preference analysis.
2. Calculation of points of interest, and related gamification information on the various datasets.
3. Processing the route through a routing engine.
4. Presentation of the proposed route to the user and activation of the crowd-sourcing mode.
5. Post-processing of data received by the user and reintegration into the dataset.

2.1 Request Phase

Through applications (mobile and web-based ones) based on our architecture, the user can make a request for a generic path from a starting point S to a destination D. The user, as we will later show in the following section, has two possible choices. S/He can specify preferences for points of interest s/he wants to visit, or let the layer customization use preferences already stored in her/his profile.

In fact, our architecture follows the MyData Philosophy as regards the management of personal information of users [15]. MyData is a technology born in Finland whose purpose is to create an efficient and above all safe solution for the management and storage of personal data. The basic idea is to have a personal dataset called Personal Data Vault (hereafter PDV), where all the personal data of a user are properly stored. However, this dataset is accessed through a specific request, previously authorized, specifying needed data and format, so that the requesting service is not provided with rough and sensitive data, but only with the most appropriate result from a suitable data aggregation and processing algorithm. More details regarding this technology can be found in [16,17].

The Preferences layer (as shown in Fig. 1) then supports an authorized MyData engine, which, if no preference is specified, will be invoked to have the corresponding user profile. However, if the user specifies some preferences, (i) they are processed in a general way by the layer and then (ii) they are possibly integrated with the personal profile from the PDV. The purpose of these two phases is to generate and produce a single profile-request where the preferences are specified according to the format we use. This pre-processing step will then serve to calculate the various categories of preference and the weight to associate with each of them.

2.2 Points of Interest Search

Once the user's preferences are processed, the layer searches for the points of interest that match the request. It is necessary at this stage to better identify the data sources. Points of interest are calculated by invoking multiple data sources for several reasons. As stated earlier, the initial purpose of this project is to provide an architecture that can handle user's preferences, by driving him/her, within a generic urban area, among the monuments and other important points of interest in terms of cultural heritage.

We have specifically taken into account two main different types of data sources: official (coming from local entities and public administrations, such as municipalities, regions, ministries, departments, tourist offices, and so on) and unofficial ones (collected by means of crowdsourcing and crowdsensing activities, involving citizens and tourists).

In particular, as regards the official data sources, we have drawn our attention to the available resources related to the cultural heritage of the Emilia-Romagna Region (where Bologna is placed) as a case study. Thus, our system can communicate with the following official and authoritative data sources:

– the SACHER Project (http://www.eng.sacherproject.com/);
– Emilia-Romagna Cultural Heritage Open Data project (http://www.patrimonioculturale-er.it/webgis/);
– Bologna Open Map (http://dati.comune.bologna.it/bolognaopenmap?language=en);
– Google Maps.
– Open Street Map.

The DB wrapper is able to communicate with this Data Sources and provide a JSON structure, which is quite simple and includes the information needed to categorize monuments, palaces, museums or other points of interest related to cultural heritage, together with information about their geographical position and geo-referenced data. A fragment of the JSON code obtained from the search query, where geo-references are categorized by the type of monuments (according to the user's preferences) can be found in http://www.cs.unibo.it/~mirri/cicerone/json1.html.

Again, in this phase we have a twofold purpose. The first goal is obviously to look for points of interest, and this is accomplished by means of a deep search through all the datasets where the results are unified, eliminating duplicates and errors. Secondly, the list of corresponding matching POIs is analyzed, so as to identity those ones with missing metadata, related media and multimedia, additional relevant information.

This does not concern the geo-reference features, but documentation related to the specific point of interest and it includes data such as photos, timetables, accessibility for people with disabilities, real-time crowding, partial closures, etc.

The resulting POIs are indeed ordered and marked based on the missing features, so that they are recognizable once the path is calculated. At each missing feature is assigned a score that will be used on the fourth stage, so as to start the gamification phase. The final result is similar to the one shown at http://www.cs.unibo.it/~mirri/cicerone/json2.html. At this stage, the list of matching points of interest is ready to be exploited as an input to the route planning engine.

2.3 Routing Trip Elaboration

The bottom level planner depicted in Fig. 1 is an engine implementing routing algorithms that, given a geo-refined map and a possible additional data set (such as GTFS public transport data), computes a path from a starting point S to a destination D, going across or through all points of interest resulting from the previous step.

Any routing engine can be exploited for this step, in particular, we have used an OpenTripPlanner (OTP) instance in the prototype we have developed. The result of the OTP processing is another JSON file (a fragment is shown in http://www.cs.unibo.it/~mirri/cicerone/json3.html). It is worth noting that, at each point of interest, the necessary intermediate points are added to graphically map the path in the next step. Note that for each point, the corresponding ID is stored in such a way that it can memorize the association between the point and the information calculated in the previous steps, both in terms of interesting features of the POI, and of lack thereof (to involve the user in providing them).

2.4 User Trip

In the fourth step, the path is ready to be exploited by the user, following all the references made in the previous phases. At this stage, there is a first update of user's preferences within the corresponding PDV. Specifically, the computed path is saved and stored, together with all the relevant marks on the points of interest required and calculated. It is important to make a first update at this stage, rather than at the next ones because it is not certain whether the user will (want to) interact later.

The path is represented on a map (typically by using a mobile application or a web-based service, just like Google Maps or Open Street Maps); each POI with some missing features is marked and shown in the map so as to let the user be aware of the possibility of earning points by providing some of the missing information. Each feature then instantiates a listener, which basically his a web hook that starts and wait for possible interaction from the user. This interaction includes several possible different actions as:

- Add/Update a description or a previous one.
- Add/Update a picture or general media information or a previous one.
- Submit a review.
- Report a problem.

2.5 Crowdsourcing Re-feed

At this stage, the user can enjoy the required path with the relevant points of interest, add the missing data from the recommended points of interest, earn scores and gain bonus points. These data are again intercepted by the preference layer and will be used in two ways. Firstly, user's preferences and history will be updated a second time within the corresponding PDV. Secondly, the layer will feed the dataset by updating the new data derived from the user's interaction.

3 A Prototype

In order to prove our architectural approach and to conduct some preliminary tests, we developed a prototype web-app, which is designed to be exploited by web browsers and mobile devices. Such a web-app (named Cicerone) has been designed with the aim of supporting tourists, citizens, and visitors in wandering around the city of Bologna, across the points of interest related to cultural heritage, showing their details, and in expressing their preferences and personalizing their routes. It has been structured in back-end (developed by using NodeJS and MongoDB) and front-end (developed by using Vue.js, SASS, CSS3 Flexbox), as a responsive Web application. A screenshot of Cicerone user interface is depicted in Fig. 2.

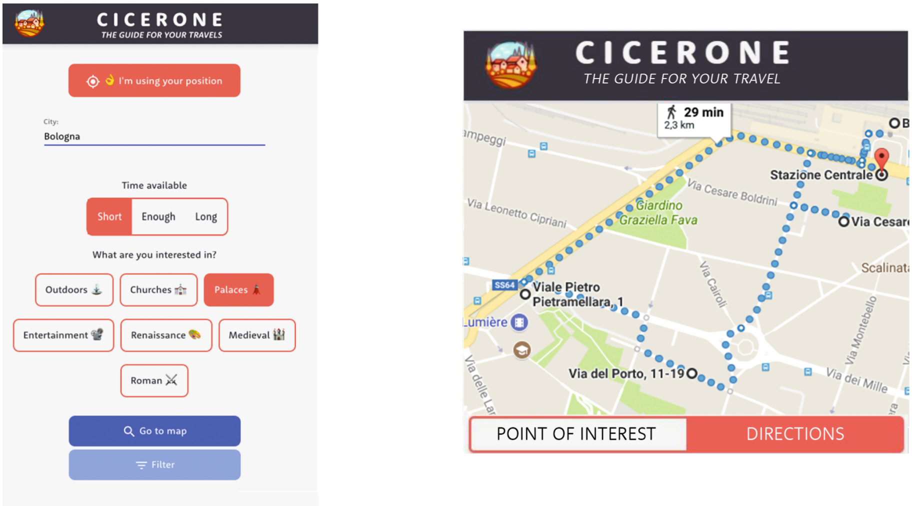

Fig. 2. A screenshot from Cicerone (left), John's path proposed by Cicerone (right)

On the basis of the described preference architecture, we have modified Cicerone app in order to be able to interact with such a preference layer. By means of Cicerone interface, the user can set some preferences in terms of time frame to spend during the visit, categories of favorite points of interests (i.e. type of monuments, ages, styles, etc.), starting position and destination of the path.

3.1 Personas and Scenarios

This subsection presents two personas and related scenarios we have defined with the aim of testing our prototype in the city of Bologna.

John stops over in Bologna during his travel by train from Venice to Florence. He has to wait 2 h for his connection, so he decides to use Cicerone to plan a small visit in Bologna.

As shown in Fig. 2, he uses the geolocation function, chooses "Short" in the time available radio button and selects that he's interested in "Palaces" then clicks to "Go to map". Cicerone proposes the following list of palaces to visit: (i) Palazzo delle Poste, (ii) Ex Conservatorio Margherita di Savoia, (iii) Ex Albergo Felsina, (iv) Palazzo Cantelli-Forti, (iv) Palazzo dell'Istituto di Aiuto Materno e di Assistenza ai Lattanti and (v) Ex Sede del Gruppo Rionale Fascista "Tabanelli". The proposed visit, as shown in Fig. 2, takes about 30 min and starts, and ends, at Bologna train station. As many of the palaces are private and do not contain museums or exhibitions, Cicerone doesn't consider a visit time for any point of interest, but it only proposes a personalized pedestrian path for such points [18]. During the path, Cicerone asks John to add some pictures of Palazzo delle Poste, since there is no photo available on the sources datasets.

Tomorrow morning, Jane will have a business meeting in Bologna, but she will have the afternoon off. She decides to have a walk to enjoy some of the churches in Bologna old town, using "Cicerone" to plan the visit. She enters Bologna in the city's text field, choose "Enough" in the time available radio button and selects that she's interested in "Churches" then clicks on "Go to map". Cicerone proposes the following list of palaces to visit: (i) Chiesa di San Benedetto, (ii) Oratorio di San Carlo Borromeo, (iii) Ex Chiesa S. Maria del Buon Pastore, (iv) Chiesa dei SS. Filippo e Giacomo and (v) Ex Convento annesso alla Chiesa dei SS. Filippo e Giacomo. The visit proposed, as shown in Fig. 3, takes about 40 min and starts, and ends, at Bologna train station. At this time, however, the visit time for each point of interest has to be added. Actually the visit time is the same for each point of interest in "Church" category. During the visit Cicerone notifies some tags that can be added to the churches (e.g. accessibility for people with

Fig. 3. Jane's path proposed by Cicerone

disabilities, availability of guided tours and visits, availability of wifi connection, availability of toilettes, religious services, services for babies, etc.). Cicerone lets the user also add information about the specific time devoted to the visit of each specific point of interest, enriching the whole system and improving the personalization settings and the computations related to the personalized routes.

4 Conclusions

In this paper, we presented a system, based on a microservice architecture, devoted to let the user set specific preferences and settings and to compute personalized routes across cultural heritage in a urban environment. The system integrates many and different data sources, mainly based on open data, provided by official entities (such as public and local administrations, e.g. municipalities, touristic offices, regions, etc.).

We are investigating crowdsourcing and gamification mechanisms that will enrich our system, with the aim of engaging users in providing and sharing additional data and resources related to the cultural heritage they meet during their routes. Thus, this turns citizens, visitors, and tourists into a kind of unofficial sources of information, which will be integrated with the official ones, improving the amount of available resources and supporting knowledge sharing activities. Another future work is based on the distribution of the proposed architecture in a cloud based one, by integrating in a deeper way the urban infrastructure [19], falling under the hat of Software as a Service (SaaS).

Acknowledgements. The authors want to thank Michele Campobasso, Alberto Giunta and Lisa Mazzini (who developed the first prototype of Cicerone), together with Francesca Galanti and Catia Prandi for their precious support and significant collaboration.

References

1. Marfia, G., Pau, G., Giordano, E., De Sena, E., Gerla, M.: VANET: on mobility scenarios and urban infrastructure. A case study. In: 2007 Mobile Networking for Vehicular Environments, pp. 31–36. IEEE (2007)
2. Mirri, S., Prandini, M., Salomoni, P., Delnevo, G., Callegati, F., Melis, A.: I want to ride my bicycle: a microservice-based use case for a MaaS architecture. In: 2017 IEEE Symposium on Computers and Communication (ISCC). IEEE (2017)
3. Bujari, A., Ciman, M., Gaggi, O., Palazzi, C.E.: Using gamification to discover cultural heritage locations from geo-tagged photos. Pers. Ubiquitous Comput. **21**(2), 235–252 (2017)
4. Dusi, N., Ferretti, I., Furini, M.: A transmedia storytelling system to transform recorded film memories into visual history. Entertain. Comput. **21**, 65–75 (2017)
5. Dusi, N., Ferretti, I., Furini, M.: PlayTheCityRE: a visual storytelling system that transforms recorded film memories into visual history. In: 2016 IEEE Symposium on Computers and Communication (ISCC), pp. 85–90, June 2016

6. Roccetti, M., Marfia, G., Palazzi, C.E.: Entertainment beyond divertissment: using computer games for city road accessibility. Comput. Entertain. (CIE) **9**(2), 10 (2011)
7. Oomen, J., Aroyo, L.: Crowdsourcing in the cultural heritage domain: opportunities and challenges. In: Proceedings of the 5th International Conference on Communities and Technologies, pp. 138–149. ACM (2011)
8. Stathopoulou, E., Georgopoulos, A., Panagiotopoulos, G., Kaliampakos, D.: Crowdsourcing lost cultural heritage. ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci. **2**(5), 295 (2015)
9. Melis, A., Mirri, S., Prandi, C., Prandini, M., Salomoni, P., Callegati, F.: Integrating personalized and accessible itineraries in MaaS ecosystems through microservices. Mob. Netw. Appl. 1–10 (2017)
10. Mirri, S., Prandi, C., Salomoni, P., Callegati, F., Melis, A., Prandini, M.: A service-oriented approach to crowdsensing for accessible smart mobility scenarios. Mob. Inf. Syst. **2016** (2016)
11. Callegati, F., Gabbrielli, M., Giallorenzo, S., Melis, A., Prandini, M.: Smart mobility for all - a global federated market for mobility-as-a-service operators. In: 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), pp. 1–8. IEEE (2017)
12. Newman, S.: Building Microservices. O'Reilly Media, Inc., Newton(2015)
13. Dragoni, N., Giallorenzo, S., Lluch-Lafuente, A., Mazzara, M., Montesi, F., Mustafin, R., Safina, L.: Microservices: yesterday, today, and tomorrow. CoRR, vol. abs/1606.04036 (2016)
14. Lea, G.: Microservices security: all the questions you should be asking (2015)
15. Kuikkaniemi, K., Poikola, A., Honko, H.: Mydata a nordic model for human-centered personal data management and processing. Technical report, Ministry of Transport Finland (2010)
16. Poikola, A., Kuikkaniemi, K., Honko, H.: Mydata - white paper. Technical report, November 2016
17. "Mydata 2016. Advancing human centric personal data," November 2016
18. Mirri, S., Prandi, C., Salomoni, P.: Personalizing pedestrian accessible way-finding with mpass. In: 2016 13th IEEE Annual Consumer Communications & Networking Conference (CCNC), pp. 1119–1124. IEEE (2016)
19. Callegati, F., Cerroni, W., Contoli, C., Santandrea, G.: Implementing dynamic chaining of virtual network functions in openstack platform. In: 2015 17th International Conference on Transparent Optical Networks (ICTON), pp. 1–4. IEEE (2015)